WHAT IS CLAIMED IS:

1. A programmable processing system comprising:

a first processor for executing a first portion of an instruction; and

a second processor for executing a second portion of the instruction, wherein the second portion of the instruction is interpreted by the first processor as an extension to an immediate operand field included in the first portion of the instruction to define a larger immediate operand field.

2. The system of claim 1, wherein before the second portion of the instruction is interpreted by the first processor a previous instruction is interpreted by the second processor that indicates a number of instructions after the previous instruction that may be interpreted by the first processor as an extension to the immediate operand field.

3. The system of claim 2 further comprising:

a device for storing the number of instructions and decrementing the number stored with the number being decremented once for each instruction executed by the first processor.

4. The system of claim 3 further comprising:

decode logic to decode control fields of the instructions for the second processor, with the decoding logic interpreting the previous instruction and storing the number of instructions indicated by the instruction in the register.

5. The system of claim 4, wherein said decoding logic receives the stored register value and outputs the second portion of the instruction to the first processor if the stored value is not zero.

6. The system of claim 5, wherein said decoding logic decodes the second portion of the instruction and controls the second processor if the stored value is zero.

7. The system of claim 6, wherein said first processor and said second processor execute multiple contexts.

8. The system of claim 7 further comprises:

an execution control stack that includes a storage area for the remaining number of instructions for at least one context.

9. The system of claim 5, wherein the decrementing circuit further comprises:

an OR logic block that receives the stored register value for the current context and outputs a signal to the second processor to indicate if the current instruction may be interpreted as a long immediate extension by the first processor if any of the bits of the register value are equal to one.

10. A computer program product store on a computer readable medium comprising instructions for causing a computer to:

interpret a first portion of an instruction by a first processor; and

interpret a second portion of the instruction by a second processor as an extension to an immediate operand field included in the first portion of the instruction to define a larger immediate field.

11. The computer program product of claim 10 further comprises a previous instruction for causing a computer to:

interpret a second portion of said previous instruction by the second processor that indicates a number of

instructions which may thereafter be interpreted by the first processor as an extension to the immediate operand field of the first processor instruction.

12. The computer program product of claim 11 wherein interpreting the second portion of said previous instruction causes the number of instructions to be stored in a register.

13. The computer program product of claim 12 wherein said instructions causing a computer to interpret the second portion of the instruction further comprises instructions causing a computer to:

decrement the stored value in the register.

14. The computer program product of claim 13 wherein said instructions causing a computer to interpret a second portion of the instruction by a second processor as an extension to an immediate operand field further comprises instructions causing a computer to:

send the second portion of the instruction to the first processor if the stored value in the register is not zero.

15. The computer program product of claim 12 wherein said instructions causing a computer to interpret the second portion of the instruction further comprises instructions causing a computer to:

OR the bits stored in the register; and

Send a signal to a decode logic for the second processor indicating that a bit was set in the register.

16. The computer program product of claim 12 wherein interpreting the second portion of said previous instruction causes the number of instructions to be stored in an execution control stack.

17. A programmable processing system comprising:

a first processor for executing a first portion of an instruction; and

a second processor for executing a second portion of the instruction,

wherein the second portion of the instruction is interpreted by the first processor as an extension to an immediate operand field included in the first portion of the instruction to define a larger immediate operand field, and

wherein before the second portion of the instruction is interpreted by the first processor a previous instruction is interpreted by the second processor to enable the extension to the immediate operand field.

18. The system of claim 17 further comprising:

decode logic to decode control fields of the instructions for the second processor, with the decoding logic interpreting the previous instruction and setting a bit to indicate the extension to the immediate operand field.

19. The system of claim 18, wherein said decoding logic outputs the second portion of the instruction to the first processor if the bit is set.

20. The system of claim 19, wherein said decoding logic decodes the second portion of the instruction and controls the second processor if the stored bit is cleared.

21. The system of claim 20, wherein said first processor and said second processor execute multiple contexts.

22. The system of claim 21 further comprises:

an execution control stack that includes a storage area for the remaining number of instructions for at least one context.

23. A computer program product store on a computer readable medium comprising instructions for causing a computer to:

interpret a first portion of an instruction by a first processor;

interpret a second portion of the instruction by a second processor as an extension to an immediate operand field included in the first portion of the instruction to define a larger immediate field; and

interpret a second portion of said previous instruction by the second processor that indicates instructions may thereafter be interpreted by the first processor as an extension to the immediate operand field of the first processor instruction.

24. The computer program product of claim 23 wherein interpreting the second portion of said previous instruction causes the causes the setting of a bit.

25. The computer program product of claim 24 wherein said instructions causing a computer to interpret a second portion of the instruction by a second processor as an extension to an immediate operand field further comprises instructions causing a computer to:

send the second portion of the instruction to the first processor if the stored bit is set.